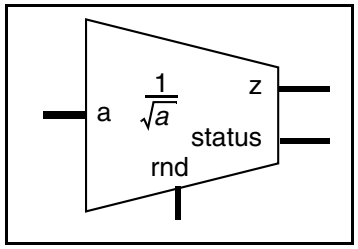
DW_fp_invsqrt

Floating-Point Reciprocal of Square Root

Version, STAR, and myDesignWare Subscriptions: IP Directory

Features and Benefits

Revision History



- The floating-point format is controlled by parameters, and covers formats in the IEEE 754 standard
- Exponents can range from 3 to 31 bits
- Fractional part of the floating-point number can range from 2 to 253 bits
- A parameter controls the use of denormal values
- Provides a variety of rounding modes
- Accuracy conforms to IEEE 754 standard
- DesignWare datapath generator is employed for better timing and area

Description

DW_fp_invsqrt is a component that works with floating-point values to compute the reciprocal of the square-root of a floating-point input a, to produce a floating-point result $1/\sqrt{a}$. The output of this component has accuracy consistent with the IEEE Standard 754.

Component pins are described in Table 1-1 and configuration parameters are described in Table 1-2.

Table 1-1 Pin Description

Pin Name	Width	Direction	Function
a	(*sig_width* + *exp_width* + 1) bits	Input	FP Input data
rnd	3 bits	Input	Rounding mode; supports all rounding modes described in the *Datapath Floating-Point Overview*
z	(*sig_width* + *exp_width* + 1) bits	Output	FP output data
status	8 bits	Output	Status flags for result `z` For details, see STATUS Flags in the *Datapath Floating-Point Overview*.

Table 1-2 Parameter Description

Parameter	Values	Description
sig_width	2 to 253 bits	Word length of fraction field of floating-point numbers `a` and `z`
exp_width	3 to 31 bits	Word length of biased exponent of floating-point numbers `a` and `z`

Table 1-2 Parameter Description (Continued)

Parameter	Values	Description
ieee_compliance	0 or 1	Level of support for IEEE 754: ■ 0: No support for IEEE 754 NaNs and denormals; NaNs are considered infinities and denormals are considered zeros ■ 1: Fully compliant with the IEEE 754 standard, including support for NaNs and denormals For more, see IEEE 754 Compatibility in the *Datapath Floating-Point Overview*.

Table 1-3 Synthesis Implementations

Implementation Name	Function	License Feature Required
rtl	Synthesis model	DesignWare

Table 1-4 Simulation Models

Model	Function
DW02.DW_FP_INVSQRT_CFG_SIM	Design unit name for VHDL simulation
dw/dw02/src/DW_fp_invsqrt_sim.vhd	VHDL simulation model source code
dw/sim_ver/DW_fp_invsqrt.v	Verilog simulation model source code

Table 1-5 Functional Description

a	status	z[a]
a (FP)	*	z = $1/\sqrt{a}$

a. The actual output value is defined by the rounding mode

Alternative Implementation of Floating-point Reciprocal Square Root with DW_lp_fp_multifunc

The floating-point reciprocal square root operation can also be implemented by DW_lp_fp_multifunc component, which evaluates the value of floating-point reciprocal square root with 1 ulp error bound. There will be 1 ulp difference between the value from DW_lp_fp_multifunc and the value from DW_fp_invsqrt. Performance and area of the synthesis results are different between the DW_fp_invsqrt and reciprocal square root implementation of the DW_lp_fp_multifunc, depending on synthesis constraints, library cells and synthesis environments. By comparing performance and area between the reciprocal square root implementation of DW_lp_fp_multifunc and DW_fp_invsqrt component, the DW_lp_fp_multifunc provides more choices for the better synthesis results.

Below is an example of the Verilog description for the floating-point reciprocal square root of the DW_lp_fp_multifunc. For more detailed information, see the DW_lp_fp_multifunc datasheet.

```
DW_lp_fp_multifunc #(sig_width, exp_width, ieee_compliance, 4) U1 (
    .a(a),
    .func(16'h0004),
    .rnd(3'h0),
    .z(z),
    .status(status)
);
```

For more information on the floating-point system defined for all the DesignWare Floating-point components, including `status` flag bits, floating-point formats and compatibility with IEEE standard, see the *Datapath Floating-Point Overview*.

Suppressing Warning Messages During Verilog Simulation

The Verilog simulation model includes macros that allow you to suppress warning messages during simulation.

To suppress all warning messages for all DWBB components, define the DW_SUPPRESS_WARN macro in either of the following ways:

- Specify the Verilog preprocessing macro in Verilog code:

  ```
  `define DW_SUPPRESS_WARN
  ```

- Or, include a command line option to the simulator, such as:

  `+define+DW_SUPPRESS_WARN` (which is used for the Synopsys VCS simulator)

The warning messages for this model include the following:

- If an invalid rounding mode has been detected on `rnd`, the following message is displayed:

  ```
  WARNING: <instance_path>:
      at time = <timestamp>: Illegal rounding mode.
  ```

  To suppress this message, use the DW_SUPPRESS_WARN macro explained earlier.

Related Topics

- Datapath Floating-Point Overview
- DesignWare Building Block IP User Guide

HDL Usage Through Component Instantiation - VHDL

```
library IEEE,DWARE;
use IEEE.std_logic_1164.all;
use DWARE.DW_Foundation_comp.all;

entity DW_fp_invsqrt_inst is
      generic (
        inst_sig_width : POSITIVE := 23;
        inst_exp_width : POSITIVE := 8;
        inst_ieee_compliance : INTEGER := 0
        );
      port (
        inst_a : in std_logic_vector(inst_sig_width+inst_exp_width downto 0);
        inst_rnd : in std_logic_vector(2 downto 0);
        z_inst : out std_logic_vector(inst_sig_width+inst_exp_width downto 0);
        status_inst : out std_logic_vector(7 downto 0)
        );
    end DW_fp_invsqrt_inst;


architecture inst of DW_fp_invsqrt_inst is

begin

    -- Instance of DW_fp_invsqrt
    U1 : DW_fp_invsqrt
    generic map ( sig_width => inst_sig_width, exp_width => inst_exp_width,
ieee_compliance => inst_ieee_compliance )
    port map ( a => inst_a, rnd => inst_rnd, z => z_inst, status => status_inst );


end inst;

-- pragma translate_off
configuration DW_fp_invsqrt_cfg_inst of DW_fp_invsqrt_inst is
for inst
end for; -- inst
end DW_fp_invsqrt_cfg_inst;
-- pragma translate_on
```

HDL Usage Through Component Instantiation - Verilog

```
module DW_fp_invsqrt_inst( inst_a, inst_rnd, z_inst, status_inst );

parameter sig_width = 23;
parameter exp_width = 8;
parameter ieee_compliance = 0;


input [sig_width+exp_width : 0] inst_a;
input [2 : 0] inst_rnd;
output [sig_width+exp_width : 0] z_inst;
output [7 : 0] status_inst;

    // Instance of DW_fp_invsqrt
    DW_fp_invsqrt #(sig_width, exp_width, ieee_compliance)
      U1 ( .a(inst_a), .rnd(inst_rnd), .z(z_inst), .status(status_inst) );

endmodule
```

Revision History

For notes about this release, see the *DesignWare Building Block IP Release Notes*.

For lists of both known and fixed issues for this component, refer to the STAR report.

For a version of this datasheet with visible change bars, click here.

Date	Release	Updates
July 2020	DWBB_201912.5	■ Adjusted the description of the *ieee_compliance* parameter in Table 1-2 on page 1 ■ Added "Suppressing Warning Messages During Verilog Simulation" on page 3
January 2020	DWBB_201912.1	■ Corrected port names for DW_lp_fp_multifunc in "Alternative Implementation of Floating-point Reciprocal Square Root with DW_lp_fp_multifunc" on page 3
July 2019	DWBB_201903.3	■ Removed reference to minPower library in "Alternative Implementation of Floating-point Reciprocal Square Root with DW_lp_fp_multifunc" on page 3 ■ Added this Revision History table and the document links on this page

Copyright Notice and Proprietary Information

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at https://www.synopsys.com/company/legal/trademarks-brands.html.

All other product or company names may be trademarks of their respective owners.

Free and Open-Source Software Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.
www.synopsys.com